

Nome: Joao rodrigues neto

Curso: SIN

Trabalho de estrutura de dados I

Questão 06 - Você está implementando um sistema operacional e precisa gerenciar a ordem de execução de processos e threads. Qual estrutura de dados seria mais adequada? Explique resumidamente esta estrutura.

Estrutura de Dados: Fila (Queue)

A **fila** é uma estrutura de dados linear que segue o princípio **FIFO (First In, First Out)**, ou seja, o primeiro elemento a ser inserido é o primeiro a ser removido. Essa estrutura é muito adequada para gerenciar processos e threads, pois permite que os processos sejam executados na ordem em que foram inseridos.

Como uma fila é usada no gerenciamento de processos:

- Cada processo é adicionado à fila de **pronto** quando está pronto para ser executado.
- O escalonador de processos seleciona o próximo processo da fila (o que está na frente) para ser executado.
- Após a execução, o processo é removido da fila.

Fila de Prioridades

Uma **fila de prioridades** é uma variação da fila em que cada elemento (processo ou thread) tem uma prioridade associada. Em vez de seguir estritamente a ordem de chegada, a fila de prioridades seleciona o elemento com a **maior prioridade** para ser executado primeiro.

Como uma fila de prioridades é usada no gerenciamento de processos:

- Cada processo é associado a uma prioridade, como tempo real, alta, média ou baixa.
- O escalonador escolhe o processo com a prioridade mais alta para ser executado.

- Se dois processos tiverem a mesma prioridade, o escalonador segue a ordem de chegada (FIFO).

Resumo:

- **Fila (Queue):** Permite a execução dos processos na ordem de chegada, sendo adequada para sistemas que seguem o princípio FIFO.
- **Fila de Prioridades:** Permite dar prioridade a certos processos ou threads, sendo ideal para sistemas que precisam tratar diferentes níveis de urgência.

Essa abordagem garante que o sistema operacional possa gerenciar eficientemente a ordem de execução dos processos, permitindo que processos mais importantes sejam executados antes.

Questão 07 - Você está implementando uma função em seu sistema que ao ser executada desfaz ou refaz a última ação tomada. Qual estrutura de dados seria mais adequada? Explique resumidamente esta estrutura

Estrutura de Dados: Pilha (Stack)

A **pilha** é uma estrutura de dados linear que segue o princípio **LIFO (Last In, First Out)**, ou seja, o último elemento a ser inserido é o primeiro a ser removido. Isso é exatamente o que você precisa para implementar as operações de "desfazer" (undo) e "refazer" (redo), já que a última ação realizada deve ser a primeira a ser revertida.

Como a pilha é usada no gerenciamento de "desfazer" e "refazer":

- **Desfazer (Undo):** Cada vez que uma ação é realizada, ela é armazenada no topo da pilha. Quando o usuário escolhe "desfazer", a última ação (a que está no topo da pilha) é removida e revertida.
- **Refazer (Redo):** Se uma ação é desfeita e o usuário deseja refazê-la, essa ação pode ser colocada em uma segunda pilha, chamada **pilha de redo**. Ao refazer, a ação é movida da pilha de redo de volta para a pilha de undo.

Resumo do fluxo:

1. **Ação realizada:** Armazene a ação na pilha de undo.
2. **Desfazer:** Remova a última ação da pilha de undo e coloque-a na pilha de redo.

3. **Refazer:** Remova a última ação da pilha de redo e coloque-a de volta na pilha de undo.

Vantagens da pilha para esse caso:

- **Desfazer rápido:** A pilha permite acessar e desfazer a última ação rapidamente, pois ela está sempre no topo.
- **Facilidade de implementação:** A simplicidade da estrutura LIFO facilita o gerenciamento de ações consecutivas.

Exemplo:

- Pilha de **undo**: Ação 1, Ação 2, Ação 3 → Última ação (Ação 3) é desfeita primeiro.
- Pilha de **redo**: Se a Ação 3 for desfeita, ela vai para a pilha de redo, e o usuário pode refazer a ação se desejar.

Essa abordagem é eficiente para sistemas que lidam com edição de texto, gráficos e outras tarefas onde o usuário precisa desfazer ou refazer ações sequenciais.